

معمارية الحاسوب

Architecture Computer

المحاضرة السادسة

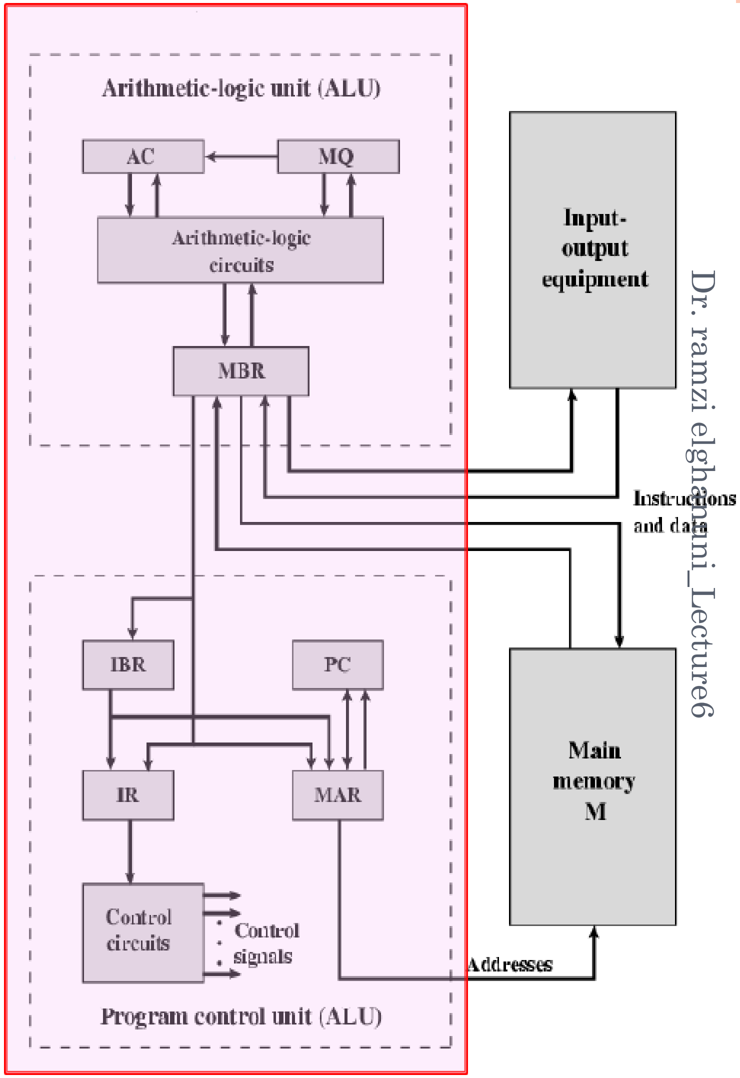
Instruction Sets: Characteristics and Functions

خصائص تعليمات الالة ووظائفها

IAS Microprocessor

المعالجات الدقيقة

- Memory Buffer Register (MBR)
- Memory Address Register (MAR)
- Instruction Register (IR)
- Instruction Buffer Register (IBR)
- Program Counter (PC)
- Accumulator (AC)
- Multiplier Quotient (MQ)



What is an Instruction Set?
ماهي مجموعة التعليمات

- مجموعة كاملة من التعليمات (Instruction) التي يمكن للمعالج (CPU) أن ينفذها.
- لغة الالة (Machine Code)
- ثنائي (Binary)
- يمثل عادة من قبل رمز التجميع Usually represented by assembly codes



Elements of an Instruction
عناصر التعليمة

- رمز العملية (Operation code)

يحدد العملية التي يتعين القيام بها علي سبيل المثال جمع ، طرح ، ضرب، ادخال/اخراج.

- مؤشر للمعامل المصدر(Source Operand reference)

أي عملية قد تنطوي علي معامل مصدر أو اكثر وهذه المعاملات هي المدخلات.

- مؤشر للمعامل الناتج (Result Operand reference)

العملية قد تسفر عن ناتج يجب أن يحفظ.

- مؤشر للتعليمة التالية (Next Instruction Reference)

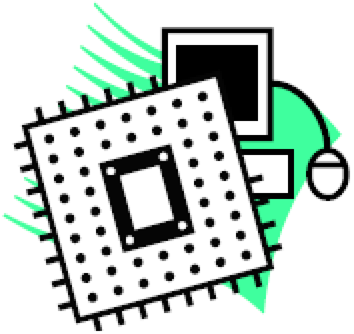
يخبر المعالج من أين تُجلب التعليمة التالية بعد إنهاء تنفيذ التعليمة الحالية.



Op Code	Source Operand	Result Operand	Next Instruction
n bits	*n* bits	*n* bits	*n* bits

Where have all the Operands Gone ?

معامل المصدر أو الناتج يمكن أن يكون ؟



- فوري (Immediate)

قيمة المعامل موجودة في حقل بالتعليمة الجاري تنفيذها.

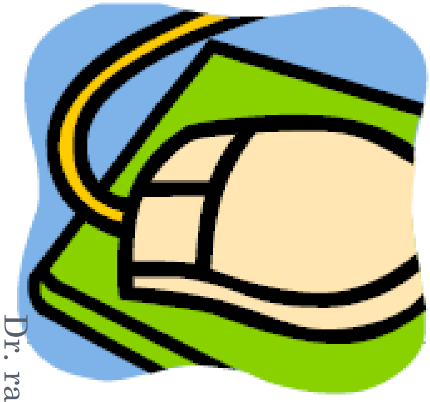
- مسجل المعالج (CPU register)

المعالج يحتوي على مسجل أو أكثر إذا كان هناك مسجل واحد فقط الاشارة إليه ضمنيا أما إذا كان أكثر من مسجل فيجب تعيين اسم أو رقم وحيد لكل مسجل والتعليمة يجب أن تحتوى على رقم أو اسم المسجل المطلوب.



Where have all the Operands Gone ?

معامل المصدرأو الناتج يمكن أن يكون ؟



- **وحدة ادخال/اخراج (I/O device)**

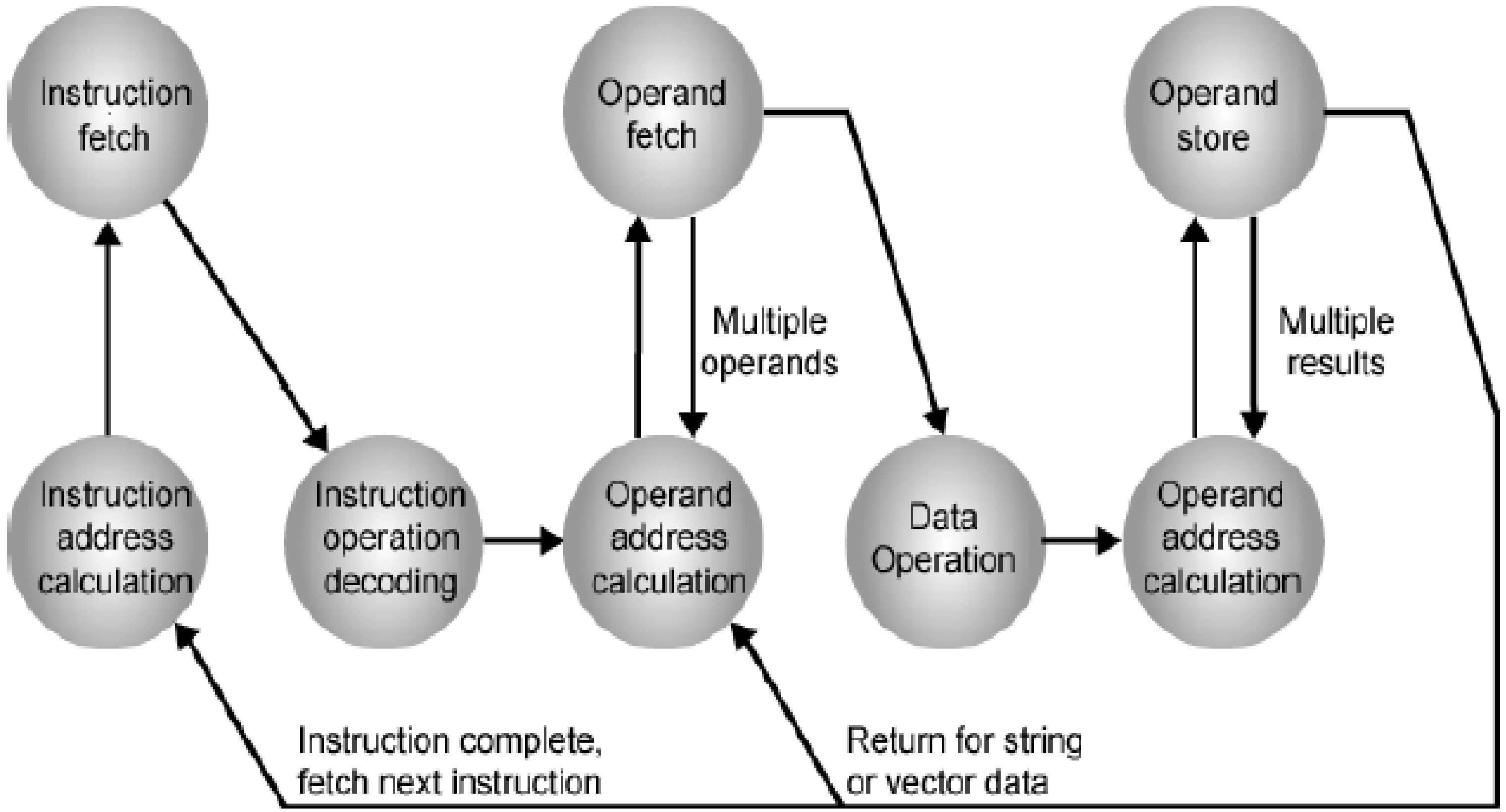
التعليمة يجب أن تحدد وحدة الادخال / الاخراج للعملية.

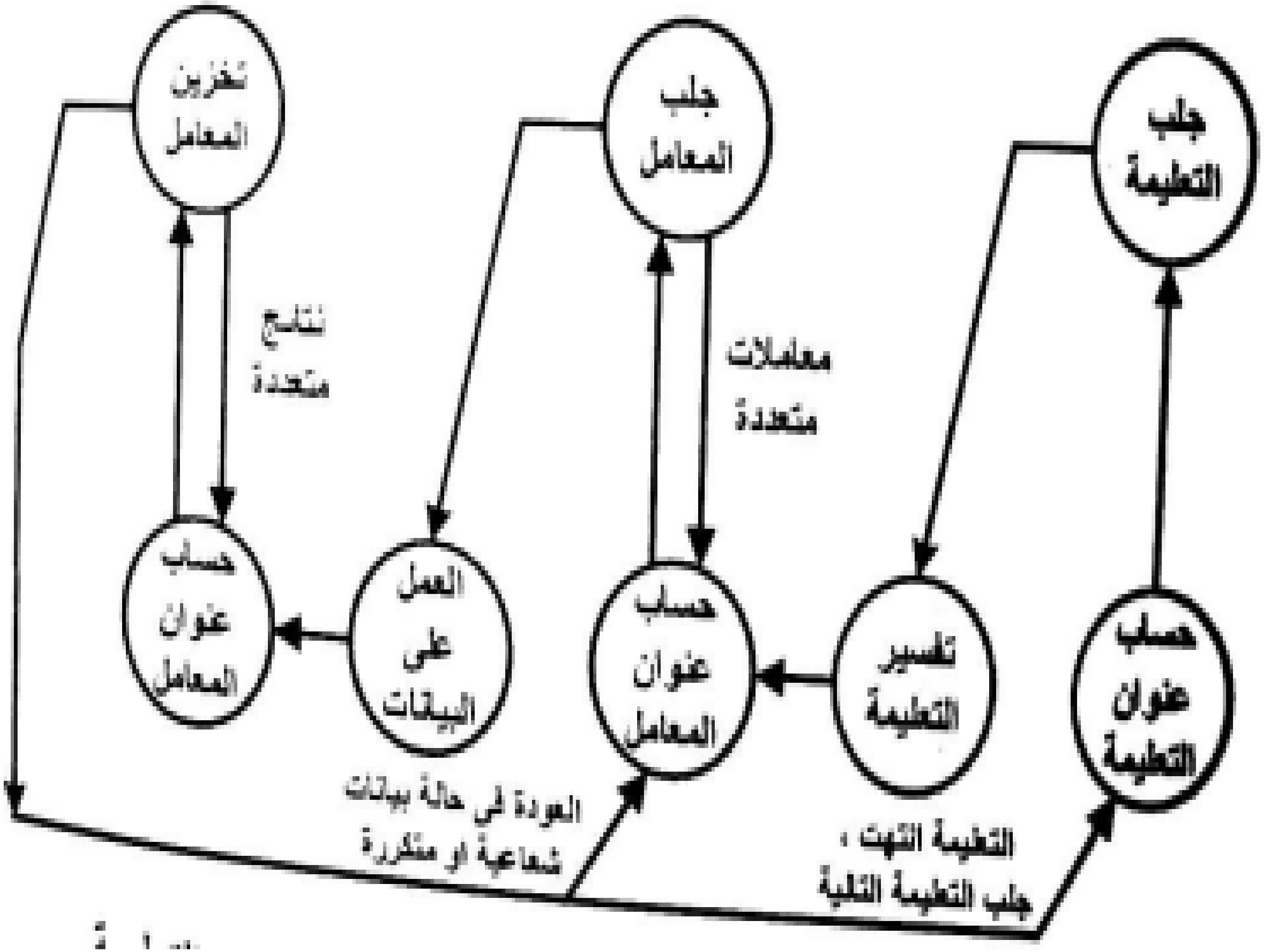
- **الذاكرة الرئيسية أو الظاهرية**

(Main memory (or virtual memory or cache))

Instruction Cycle State Diagram

مخطط حالة دورة التعليمة




تخزين المعامل
جلب المعامل
جلب التعليمة
نتائج متعددة
معاملات متعددة
حساب عنوان المعامل
العمل على البيانات
حساب عنوان المعامل
تفسير التعليمة
حساب عنوان التعليمة
العودة في حالة بيانات شعاعية او متكررة
التعليمة انتهت ، جلب التعليمة التالية

Instruction Cycle – State Diagram
دورة التعليمة – مخطط الحالة

- حساب عنوان التعليمة (IAC) تحديد عنوان التعليمة التالية اللازم تنفيذها.
- جلب التعليمة (IF) قراءة التعليمة من موقع بالذاكرة إلي المعالج .
- عملية فك تشفير التعليمة (IOD) وفهم الامر للتحديث نوع العملية والمعاملات التي تحتاجها.
- حساب عنوان المعامل (OAC) إذا كانت العملية تشتمل الاشارة إلي معامل في الذاكرة أو متاح عبر إدخال / اخراج.
- جلب المعامل (OF) من الذاكرة أو قراءتها من وحدة الادخال /الاخراج.
- تشغيل البيانات (DO) تنفيذ العملية المشار إليها في التعليمة.
- تخزين المعامل (OS) كتابة الناتج في الذاكرة أو إلي وحدة الادخال / الاخراج .

Instruction Representation

تمثيل التعليمات

من الصعب على المبرمج والقارئ على حد سواء التعامل مع التمثيل الثنائي لتعليمات المعالج.

كل تعليمة لديها تمثيل رمزي فريد من نوعه .

e.g. ADD, SUB, LOAD ،STOR , DIV , MUL

ويمكن أيضا أن المعاملات تكون ممثلة بطريقة التالية :

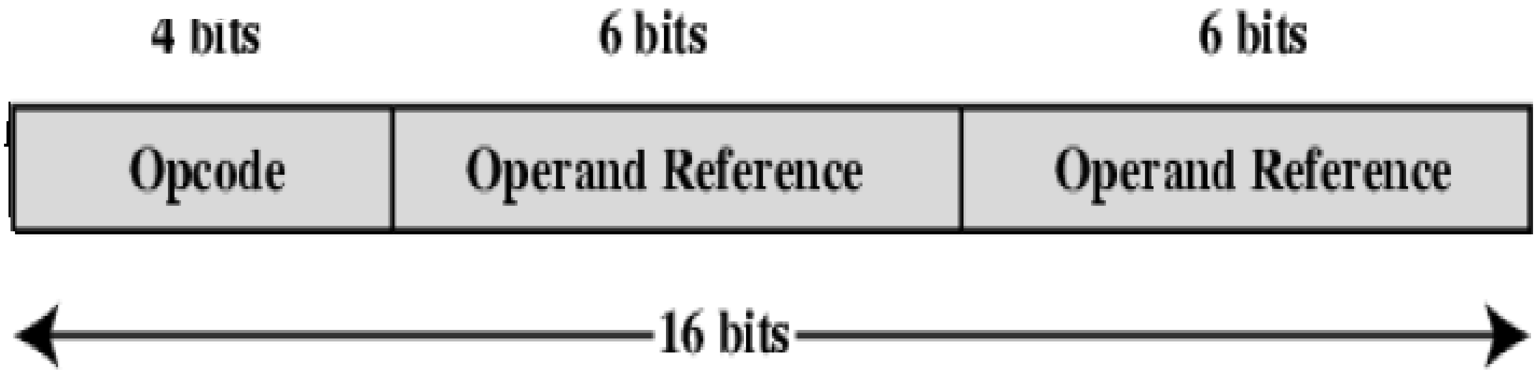
ADD A,B

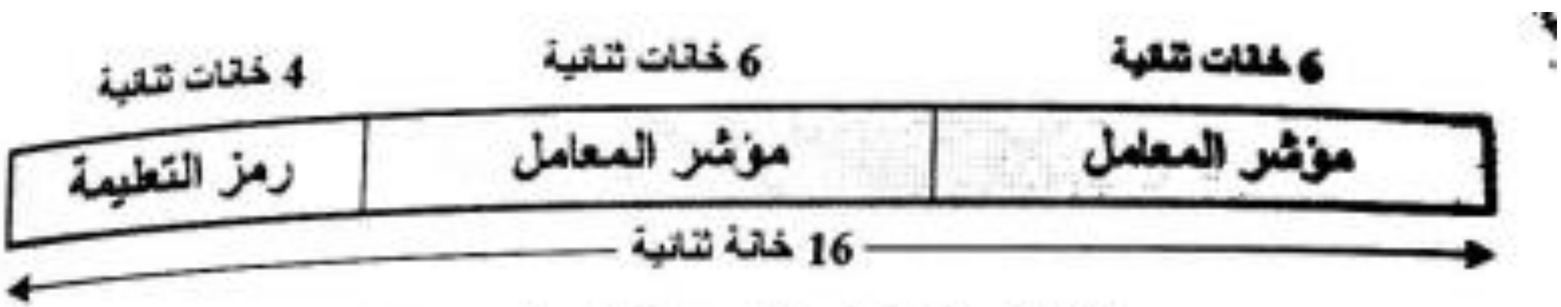
Opcode	Mnemonic	Meaning	Effect
0000	AND	Logical bit-wise AND	Rd := Rn AND Op2
0001	EOR	Logical bit-wise exclusive OR	Rd := Rn EOR Op2
0010	SUB	Subtract	Rd := Rn - Op2
0011	RSB	Reverse subtract	Rd := Op2 - Rn
0100	ADD	Add	Rd := Rn + Op2
0101	ADC	Add with carry	Rd := Rn + Op2 + C
0110	SBC	Subtract with carry	Rd := Rn - Op2 + C - 1
0111	RSC	Reverse subtract with carry	Rd := Op2 - Rn + C - 1

Simple Instruction Format

تنسيق التعليمات







الشكل (5.2) - تنسيق لتعليمة بسيطة

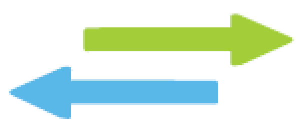
Instruction Types

أنواع التعليمات





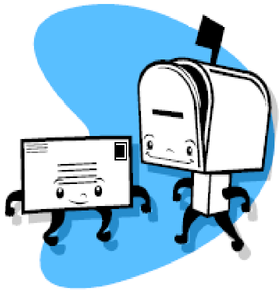
- ❖ معالجة البيانات (Data processing).
  - تعليمات الحساب والمنطق.
- ❖ تخزين البيانات (main memory) Data storage.
  - حركة البيانات إلى داخل أو خارج المسجل أو موقع بالذاكرة.



- ❖ بيانات الحركة (I/O) Data movement.
  - تعليمات الادخال / الاخراج



- ❖ التحكم (Program flow control).
  - الاختبار والتفرع.



Number of Addresses (a)

عدد العناوين

Instruction		Comment
SUB	Y, A, B	Y ← A − B
MPY	T, D, E	T ← D × E
ADD	T, T, C	T ← T + C
DIV	Y, Y, T	Y ← Y ÷ T

Y = (A − B) / [C + (D × E)]

- 3 addresses
- Operand 1, Operand 2, Result
- a = b + c;
- Not common (ليست شائعة)
- Needs very long words to hold everything

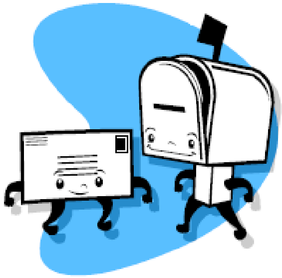
(تتطلب تنسيق تعليمة طويلة نسبيا لكى تحمل مؤشرات لثلاثة عناوين)

- May be a forth - next instruction (usually implicit)

(عنوان التعليمة التالية مشار اليها ضمنياً)

4 bits	2 bits	2 bits	2 bits
opcode	operand #1	operand #2	operand #3

ADD A,B,C (A=B+C)   1010 00 01 10



Number of Addresses (b)

عدد العناوين

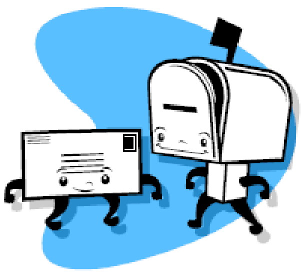
Instruction	Comment
MOVE Y, A	$Y \leftarrow A$
SUB Y, B	$Y \leftarrow Y - B$
MOVE T, D	$T \leftarrow D$
MPY T, E	$T \leftarrow T \times E$
ADD T, C	$T \leftarrow T + C$
DIV Y, T	$Y \leftarrow Y \div T$

$Y = (A - B) / [C + (D \times E)]$

- 2 addresses
- One address doubles as operand and result
- $a = a + b$
- Reduces length of instruction (يقلل من حجم طول التعليمة)
- Temporary storage to hold some results (التخزين المؤقت لعقد بعض النتائج)
- Requires some extra work (يتطلب بعض العمل الاضافي)
- لتجنب تغيير قيمة المعامل المزدوج يتم استخدام تعليمة (Move) لنقل القيم لموقع مؤقت قبل تنفيذ العملية.

4 bits	2 bits	2 bits
opcode	operand #1	operand #2

MOVE A,B	(A=B)	1000 00 01
ADD A,C	(A=A+C)	1010 00 10



Number of Addresses (c)

عدد العناوين

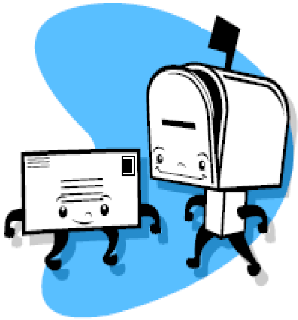
- 1 address.
- Implicit second address (العنوان الثاني ضمنياً).
- Common on early machines (منتشرعلى الأجهزة القديمة).
- Usually a register (accumulator) AC

$$Y = (A - B) / [C + (D \times E)]$$

Instruction		Comment
LOAD	D	AC ← D
MPY	E	AC ← AC × E
ADD	C	AC ← AC + C
STOR	Y	Y ← AC
LOAD	A	AC ← A
SUB	B	AC ← AC − B
DIV	Y	AC ← AC ÷ Y
STOR	Y	Y ← AC

4 bits	2 bits
opcode	operand

LOAD *B*	(*Acc*=*B*)	0000	01
ADD *C*	(*Acc*=*Acc*+*C*)	1010	10
STORE *A*	(*A*=*Acc*)	0001	00



Number of Addresses (d)

عدد العناوين

- 0 (zero) addresses
- All addresses implicit (كل العناوين ضمنياً)
- Uses a Stack (LIFO)
- e.g. push a
- push b
- add
- pop c
- Top (عنوان الموقع)
- (Top يخصص مسجل خاص لحفظ عنوان)

4 bits

opcode

PUSH *B*	(Stack=*B*)	0101
PUSH *C*	(Stack=*C*,*B*)	0110
ADD	(Stack=*B*+*C*)	1010
POP *A*	(*A*=stack)	1100

3 Addresses

$$X = (A - B + C * (D * E - F)) / (G + H * K)$$

SUB R_1,A,B	$R_1 \leftarrow A - B$
MUL R_2,D,E	$R_2 \leftarrow D * E$
SUB R_2,R_2,F	$R_2 \leftarrow R_2 - F$
MUL R_2,R_2,C	$R_2 \leftarrow R_2 * C$
ADD R_1,R_1,R_2	$R_1 \leftarrow R_1 + R_2$
MUL R_3,H,K	$R_3 \leftarrow H * K$
ADD R_3,R_3,G	$R_3 \leftarrow R_3 + G$
DIV X,R_1,R_3	$X \leftarrow R_1 / R_3$

2 Addresses

$$X = (A - B + C * (D * E - F)) / (G + H * K)$$

MOV R_1 , A	$R_1 \leftarrow A$
SUB R_1 , B	$R_1 \leftarrow R_1 - B$
MOV R_2 , D	$R_2 \leftarrow D$
MUL R_2 , E	$R_2 \leftarrow R_2 * E$
SUB R_2 , F	$R_2 \leftarrow R_2 - F$
MUL R_2 , C	$R_2 \leftarrow R_2 * C$
ADD R_1 , R_2	$R_1 \leftarrow R1 + R_2$
MOV R_3, H	$R_3 \leftarrow H$
MUL R_3 , K	$R_3 \leftarrow R_3 * K$
ADD R_3 , G	$R_3 \leftarrow R_3 + G$
DIV R_1 , R_3	$R_1 \leftarrow R_1 / R_3$
MOV X , R_1	$X \leftarrow R_1$

1 Addresses

$$X = (A - B + C * (D * E - F)) / (G + H * K)$$

LOAD A	AC ← A
SUB B	AC ← AC - B
STORE T	T ← AC
LOAD D	AC ← D
MUL E	AC ← AC * E
SUB F	AC ← AC - F
MUL C	AC ← AC * C
ADD T	AC ← AC + T
STORE T	T ← AC
LOAD H	AC ← H
MUL K	AC ← AC * K

1 Addresses

$$X = (A - B + C * (D * E - F)) / (G + H * K)$$

ADD G	AC ← AC + G
STORE T1	T1 ← AC
LOAD T	AC ← T
DIV T1	AC ← AC / T1
STORE X	X ← AC

Zero Addresses

$$X = (A - B + C * (D * E - F)) / (G + H * K)$$

PUSH A	TOS ← A
PUSH B	TOS ← B
SUB	TOS ← (A-B)
PUSH C	TOS ← C
PUSH D	TOS ← D
PUSH E	TOS ← E
MUL	TOS ← (D * E)
PUSH F	TOS ← F
SUB	TOS ← ((D*E)-F)

Number of Addresses (d)

عدد العناوين

$$X = (A - B + C * (D * E - F)) / (G + H * K)$$

MUL	TOS ← C * ((D * E) – F)
ADD	TOS ← ((A-B) + C*((D*E)-F)
PUSH G	TOS ← G
PUSH H	TOS ← H
PUSH K	TOS ← K
MUL	TOS ← (H * K)
ADD	TOS ← G + (H * K)
DIV	TOS ← ((A - B) + C * ((D * E) - F) / (G + (H * K))
POP	X ← TOS

Table 10.1 Utilization of Instruction Addresses (Nonbranching Instructions)

Number of Addresses	Symbolic Representation	Interpretation
3	OP A, B, C	A ← B OP C
2	OP A, B	A ← A OP B
1	OP A	AC ← AC OP A
0	OP	T ← (T − 1) OP T

AC = accumulator
T = top of stack
(T − 1) = second element of stack
A, B, C = memory or register locations

How Many Addresses
كم عدد العناوين

عدد العناوين لكل تعليمة (**Instruction**) هو قرار اساسي في تصميم المعالج

- **المزيد من العناوين**
  - التعليمات أكثر تعقيدا.
  - أكثر مسجلات (registers).
  - تعليمات أقل في برنامج.
  - العمليات بين المسجلات (registers) أسرع.



- **عناوين أقل**
  - معالج أقل تعقيدا.
  - تعليمات أكثر بدائية.
  - تعليمات أقل طولا.
  - عدد التعليمات تزدد في البرنامج مما يزيد من زمن التنفيذ وبالتالي برامج أطول وأكثر تعقيدا.

Design Decisions

قرارات التصميم



- مرجع العملية (Operation Repertoire)
  - ما الذي يستطيع فعله؟
  - مدى التعقيد؟
- أنواع البيانات (Data Types)
- شكل التعليمة (Instruction Formats)
  - طول حقل (Length Opcode)
  - عدد العناوين لكل تعليمة (Number of Addresses)
- المسجلات (Registers)
  - عدد من مسجلات وحدة المعالجة المركزية المتوفرة
  - العمليات التي يمكن القيام بها على المسجل؟



Types of Operand

أنواع المعاملات

<u>**تعليمات المعالج تشتغل على بيانات ، وأكثر الاصناف العامة للبيانات أهمية هي :**</u>

- **العناوين** (Addresses)
- **الارقام** (Numbers)

- عدد صحيح ثنائي أو ثنائي بنقطة ثابتة (Integer).

- ثنائي بالنقطة العائمة (floating point).

- عشري.

- **الأحرف** (Characters)

- ASCII

- **البيانات المنطقية** (Logical Data)

- Bits or flags

Types of Operation
أنواع العمليات

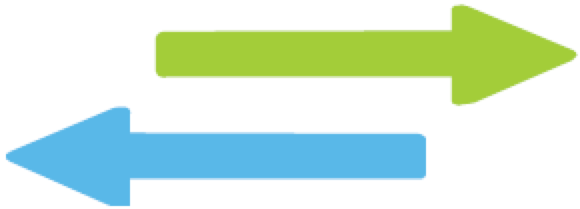
عدد رموز العمليات يختلف من معالج إلى آخر ، ومع ذلك فأنها تشترك في نفس الانواع العامة للعمليات:

- نقل البيانات (Data Transfer) .
- الحسابية (Arithmetic) .
- المنطقية (Logical) .
- التحويل (Conversion) .
- الادخال/الاخراج (I/O) .
- التحكم بالنظام (System Control) .
- نقل السيطرة (Transfer of Control) .



Data Transfer

نقل البيانات



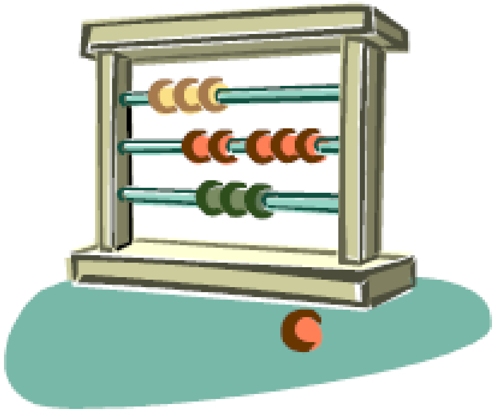
يجب على تعليمات نقل البيانات تحديد عدة أشياء:

1. يجب تحديد موقع المعامل المصدر والوجهة **(Source & Destination)** ، الموقع يمكن أن يكون في الذاكرة أو المسجل أو الجزء العلوي من المكدس **(Stack)** .

2. لا بد من الإشارة إلى طول البيانات التي سيتم نقلها.

3. يجب تحديد طريقة العنونة لكل معامل.

مثل : **Load , Store, Push , Pop , Move**



Arithmetic
الحسابية

- الجمع ، الطرح ، الضرب ، القسمة (Add, Subtract, Multiply, Divide).
- الاعداد الصحيحة بالإشارة (ذات نقطة ثابتة) (Signed Integer).
- الأرقام العشرية بالنقطة العائمة (Floating point).

وتوجد عمليات أخرى متنوعة ضمن هذا النوع من العمليات ، على سبيل المثال:

Absolute (|a|)

Increment (a++)

Decrement (a--)

Negate (-a)

Logical
المنطقية



العمليات المختصة بالبت (Bit)

AND,OR,XOR،NOT

العمليات المنطقية المختصة بالمنطق بالثنائي (Binary)

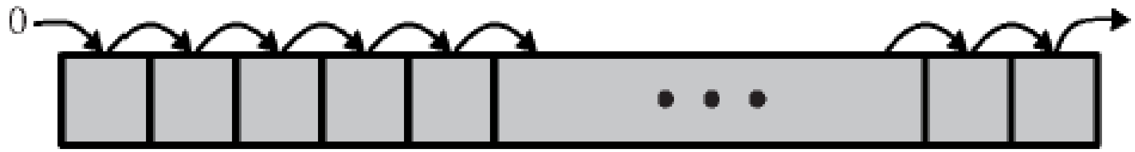
الازاحة المنطقية ، التدوير أو الازاحة الدائرية.

Shift, Rotate

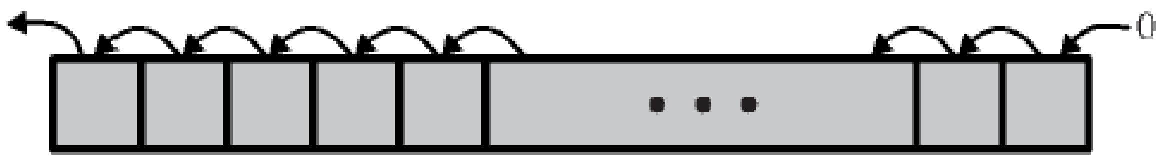
Shift and Rotate Operations

عملية الازاحة والتدوير

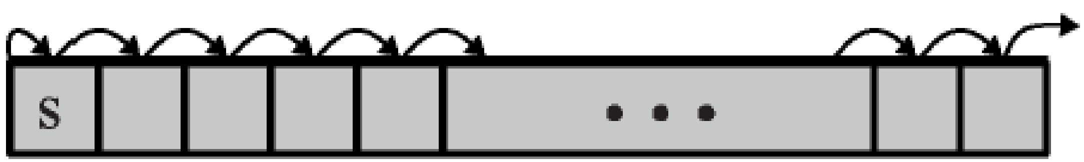




(a) Logical right shift



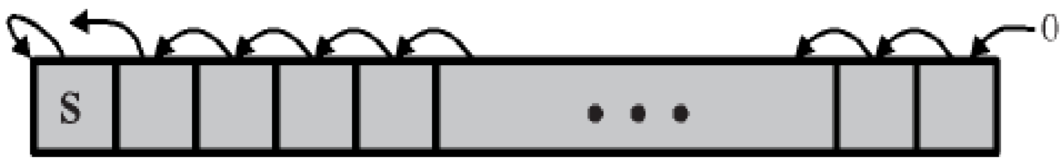
(b) Logical left shift



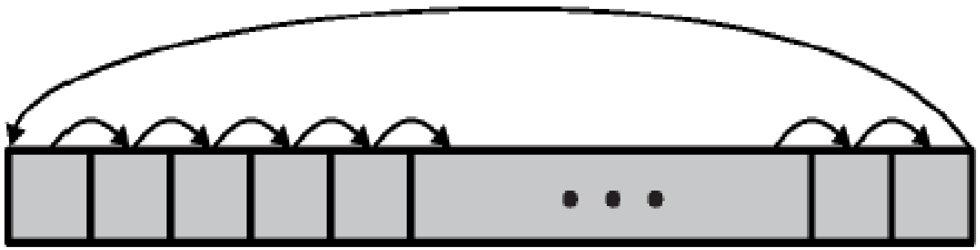
(c) Arithmetic right shift

Shift and Rotate Operations

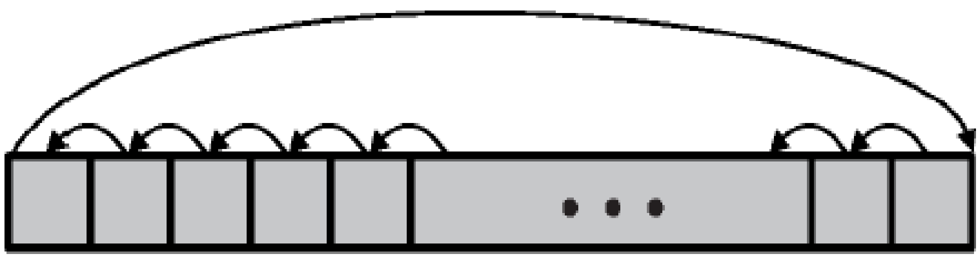
عملية الازاحة والتدوير



(d) Arithmetic left shift



(e) Right rotate



(f) Left rotate

Examples of Shift and Rotate Operations

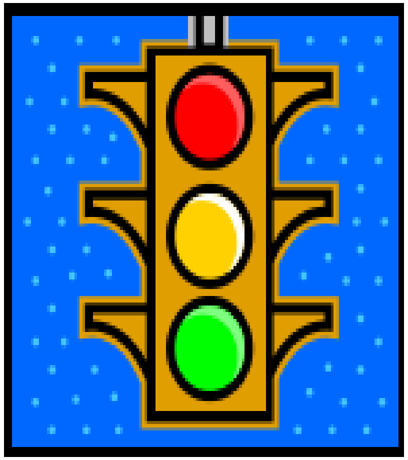
Input	Operation	Result
10100110	Logical right shift (3 bits)	00010100
10100110	Logical left shift (3 bits)	00110000
10100110	Arithmetic right shift (3 bits)	11110100
10100110	Arithmetic left shift (3 bits)	10110000
10100110	Right rotate (3 bits)	11010100
10100110	Left rotate (3 bits)	00110101

Conversion

التحويل

تعليمات التحويل هي تلك التي يمكنها تغيير تنسيق البيانات أو تعمل على تنسيق البيانات

على سبيل المثال التحويل من النظام العشري إلى النظام الثنائي.



Input / Output

الإدخال / الإخراج



- تعليمات محدودة.
- الإدخال/الإخراج المبرمج.
- الإدخال والاخراج المبرمج باستخدام الذاكرة.
- الوصول المباشر للذاكرة (DMA).
- استخدام معالج خاص بالإدخال/الإخراج .

Systems Control

التحكم بالنظام



- تعليمات التحكم بالنظام يمكن تنفيذها عندما يكون المعالج في حالة معينة وبصلاحيات خاصة.
- ينفذ برنامجا في منطقة مميزة و خاصة من الذاكرة.
- عادة ما يتم تنفيذ هذه التعليمات من قبل نظام التشغيل.

Transfer of Control
نقل السيطرة

- التفرع (**Branch**)

branch to x if result is zero -

- القفز (**Skip**)

increment and skip if zero -

ADD A Branch xxxx -

- استدعاء الإجراء (**Instruction call**)

Call and Returns -



Branch Instruction

تعليمات التفرع

التفرع يتم اذا تحقق شرط معين والا يتم تنفيذ التعليمة التالية في التسلسل (زيادة عداد البرنامج كالعادة PC) وتعليمة التفرع التي تنفذ دائما التفرع هي تفرع دون قيد أو شرط.

Branch Instruction

تعليمات التفرع

هناك طريقتين للاختبار حالة المعالج وذلك لإجراء تنفيذ تعليمة تفرع مشروط

أولا: اختبار خانة بمسجل الحالة (Status Register) الذي يعكس ناتج عملية حسابية ومنطقية يمكن أن يكون هناك أربعة أنواع مختلفة من تعليمات التفرع المشروط:

BRP X	تفرع للموقع X إذا كانت النتيجة إيجابية
BRN X	تفرع للموقع X إذا كانت النتيجة سلبية
BRZ X	تفرع للموقع X إذا كانت النتيجة صفر
BRO X	تفرع للموقع X إذا حدث فيض

في جميع الحالات المشار إليها نتيجة العملية الاخيرة التي تم تنفيذها هي التي تعيين رمز حالة (إيجابية أوسلبية أو صفر أو فيض).

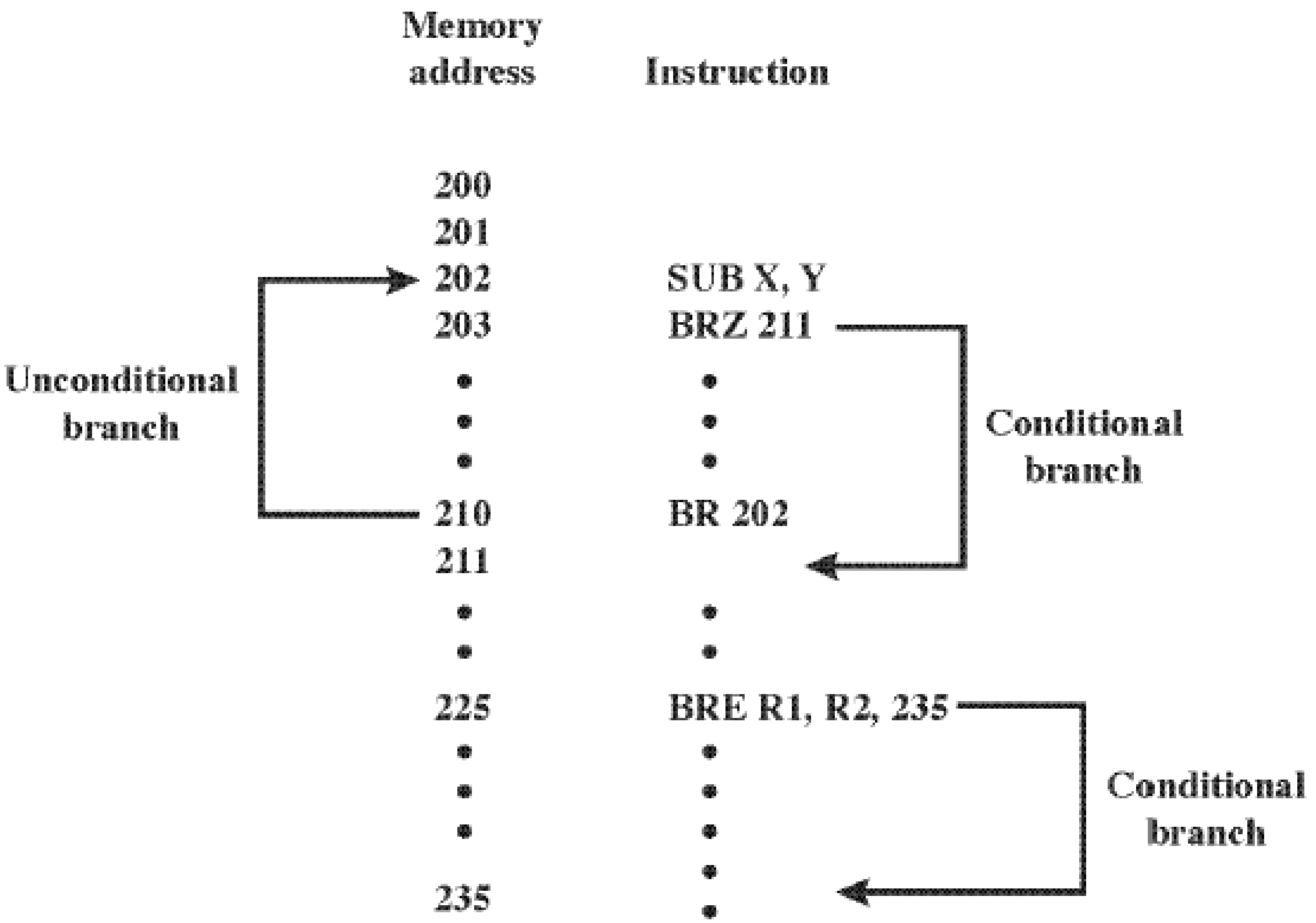
Branch Instruction

تعليمات التفرع

ثانيا : يمكن استخدام شكل من التعليمات من ثلاثة عناوين لإجراء المقارنة وتحديد التفرع في نفس التعليمة.

Branch Instruction

تعليمات التفرع



Skip Instruction
تعليمات القفز (التخطي)

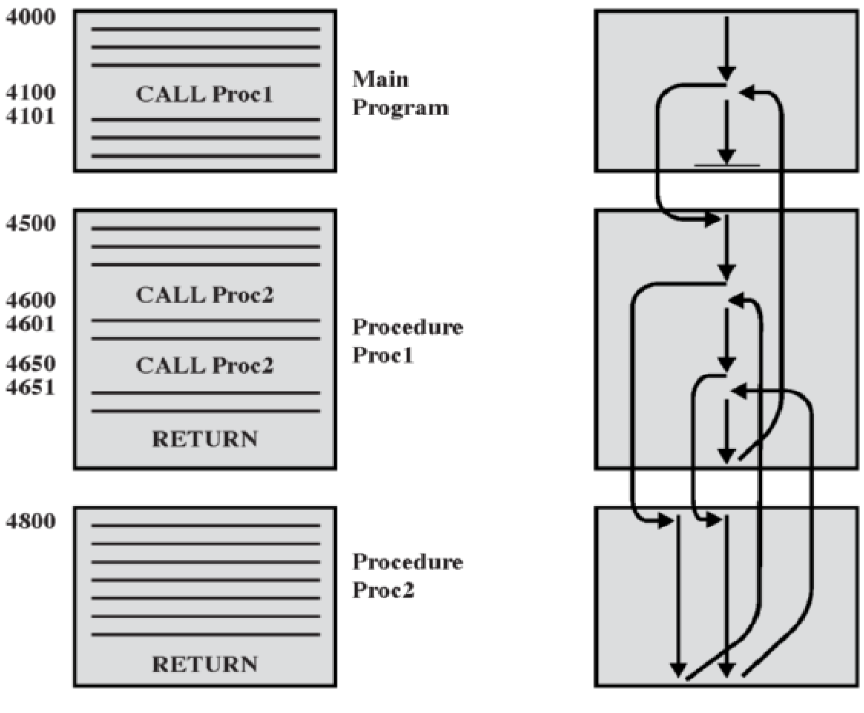
تعليمات التخطي تتضمن عنوانا ضمنياً ، والتخطي عادة يعني تخطي تعليمة واحدة ،
وبالتالي فإن العنوان الضمني يساوي عنوان التعليمة التالية مضاف إليه طول التعليمة.

Jump , Skip

Nested Procedure Calls

تعليمات استدعاء الإجراء (برنامج جزئي)

الإجراء هو نفس برنامج حاسب جزئي يتم دمجه في برنامج أوسع نطاقا. ويمكن عند أي نقطة في البرنامج استدعاء هذا الإجراء. والإيعاز للمعالج بالذهاب وتنفيذ الإجراء بأكمله من ثم العودة إلى نقطة الاستدعاء.



(a) Calls and returns (b) Execution sequence

Use of Stack

استخدام المكدس



